# NLP PROJECTS & FEEDBACKS

# FOOD2VEC  OPENFOOD FACTS

***Context***

OpenFoodFacts can be considerated as a wikipedia for food!
The goal of OpenFoodFacts is to share with everyone a maximum of informations on food products. It contains more than 2.5 millions products but maybe all products are not perfectly described... Mainly, for a product, we can find the list of ingredients, nutrition facts and food categories.

# FOOD2VEC OPENFOOD FACTS

***Expectations***

A) Vectorize list of ingredients in order to learn a word2Vec model. So we assume that the context of an ingredient is defined by the other ingredients that often occur with it. Once the model learned, you have to create a map of ingredients! To do so, reduce your embeddings to 2 dimensions (with a PCA or a UMAP reduction).

*Warning* : you will be confront to two main problems, the mistakes on the vocabulary and the size of the dataset. You can resolve this issues selecting a subset of the data. Explain and justify the bias choosen for selecting your data (if you want to manage the entire dataset, go for it, but you will need some memory and some computation time).

# FOOD2VEC  OPENFOOD FACTS

- Pretreatments
  - lowercase
  - words with more than 3 caracters
  - stopwords
  - regex : floor (15%), numbers, punctuation....
  - lemmatization/stemming
  - spellCheck
  - frequency filter

# FOOD2VEC  OPENFOOD FACTS

- Model:
  - Word2Vec (Gensim) / FastText
  - PCA, TSNE
  - UMAP

# FOOD2VEC OPENFOOD FACTS

B) Now you have vectors for ingredients, but how to use them to compare products?

Propose and implement a method allowing to compare automatically products.

With this method to evaluate Similarity between products, illustrate your approach on specific products: Select some products and show the most similar products found by your method.

# FOOD2VEC OPENFOOD FACTS

- Product representation :
  - Mean vector of ingredients
  - Keep only n first ingredients
  - Distance :
    - Euclidian distance
    - Cosinus distance
    - Weighted cosinus distance

# FOOD2VEC  OPENFOOD FACTS

C) Go forward and use your product similarity to achieve a map of products (like a Kmeans based on your product similarity for example).

# FOOD2VEC  OPENFOOD FACTS

- Kmeans (euclidian)/HDBScan(density)
- Homogeneity of clusters guided by category or nutrition facts

# PROJECT 2: INSURANCE REVIEWS

## Exploratory data analysis

Why the rating is low ? This is probably THE most important question for insurers when facing with all these reviews.

For this, you can use the two previous approaches :

- With unsupervised learning, you can find topics, and for each topic, you can give the histogram of the ratings.
- With supervised learning, you can use model interpretation (for example : https://shap.readthedocs.io/en/latest/example_notebooks/api_examples/plots/text.html)

# PROJECT 2: INSURANCE REVIEWS

- Exploratory data analysis :
  - Nan values
  - Translation with googleTrans (limitation in time and requests)
  - Translation with HuggingFace(pipeline)
  - frequency reviews by insurance company/product type...
  - distribution of ratings
  - most frequent words by company/product....
  - wordCloud

# PROJECT 2: INSURANCE REVIEWS

- Unsupervised techniques :
  - TF-IDF by score
  - LDA
  - Learning w2v / pretrained w2v
  - w2v + projection + color by rating
  - w2v + nearest words

# PROJECT 2: INSURANCE REVIEWS

- Supervised techniques :
  - Select a representation for reviews :
    - one-hot/tf-idf/meta data
    - Aggregation of embeddings/Doc2vec
    - For each review, select n embeddings of n best word according to tf-idf
- supervised machine learning : XGBoost...., regression mode
- fine-tuning bert-like model with HuggingFace


https://www.youtube.com/watch?v=V2RoqUr0qDU

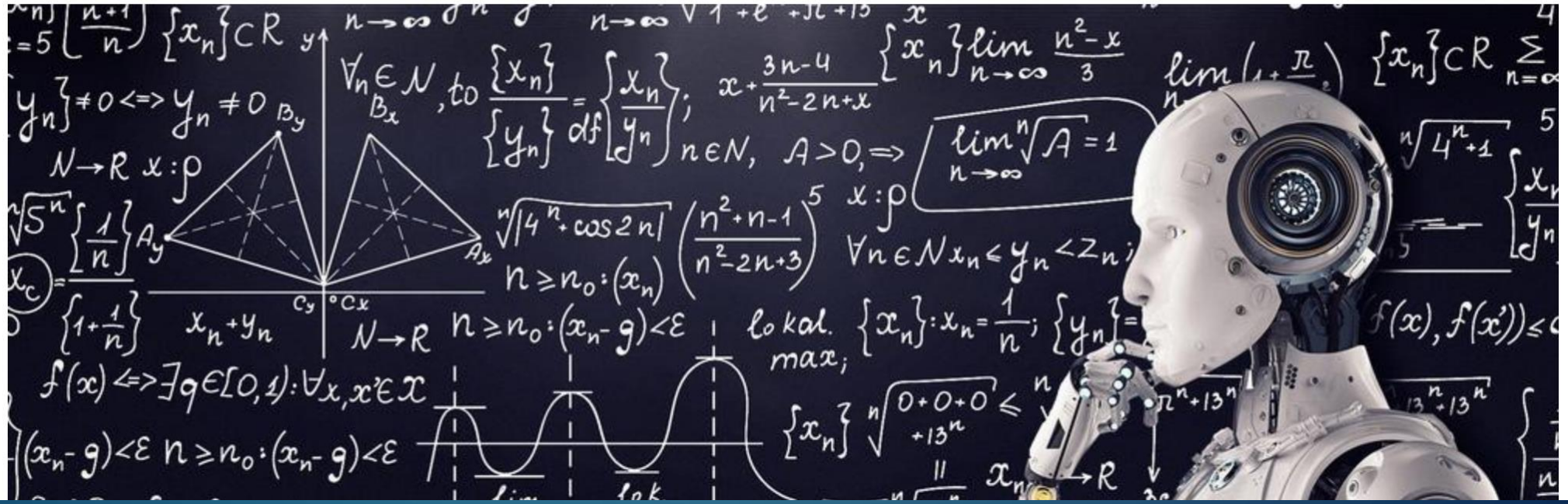
https://chat.openai.com/chat

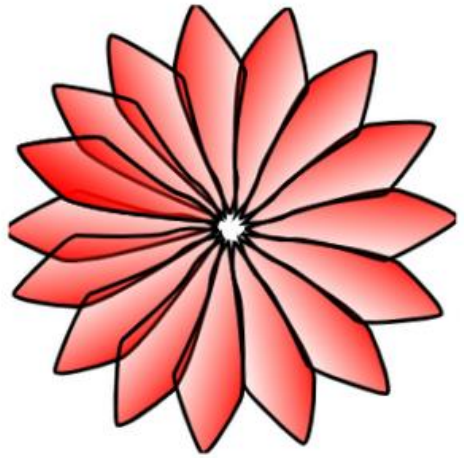
- Microsoft is preparing to add ChatGPT to Bing
  - Problem : no more visiting any website?
- Building A Virtual Machine inside ChatGPT
- Comparison to Wolfram Alpha in doing Math

# La moitié des élèves d'un master à Lyon surpris en train de tricher grâce à une intelligence artificielle

Lecture 1 min

Accueil • Sciences Et Technologie





# Petals

Run 100B+ language models at home, BitTorrent-style.
Fine-tuning and inference up to 10x faster than offloading

https://github.com/bigscience-workshop/petals